

Generating Semantically Correct Hindi Captions Using Deep Neural Network

MSc Research Project
Masters Of Science in Data Analytics

Akash Singh
Student ID:x19210736

School of Computing
National College of Ireland

Supervisor: Noel Cosgrave

National College of Ireland
Project Submission Sheet
School of Computing



Student Name:	Akash Singh
Student ID:	x19210736
Programme:	Programme Name
Year:	2021
Module:	MSc Research Project
Supervisor:	Noel Cosgrave
Submission Due Date:	16/12/2021
Project Title:	Generating Semantically Correct HindiCaptions Using Deep Neural Network
Word Count:	6003
Page Count:	19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	16th December 2021

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission , to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Generating Semantically Correct Hindi Captions Using Deep Neural Network

Akash Singh
x19210736

Abstract

Image captioning is one of the most significant and exciting challenges in computer vision and natural language processing. Several studies have been conducted in this field, the majority of which have focused on the English language. Foreign language research has a wide range of possibilities. This image captioning research is being conducted for the language Hindi. The research makes use of the Flickr-8k dataset's machine-translated Hindi captions. The research is carried out using an encoder-decoder framework. Image features are extracted using pre-trained CNNs such as VGG16, ResNet50, and Inception V3. Uni-directional and Bi-directional LSTM is employed for the text encoding process. A thorough comparison is made between various LSTM and Bi-LSTM models in this research. The VGG16 with the bi-LSTM model performed the best by giving a BLUE1 score of 0.583.

1 Introduction

1.1 Background

Image captioning is a method of creating semantically correct sentences that describe an image using computers. Describing an image is a simple process for humans, but employing computers to do so is a difficult undertaking. Because of breakthroughs in the field of neural networks, creating image captions is now a straightforward operation. There have been numerous deep learning models used to produce captions from images. One efficient method for generating these captions is to use an encoder-decoder approach. The most difficult aspect of this task is determining how accurately we describe these images. Vinyals et al. (2015) implemented the encoder-decoder model far better than earlier approaches. Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) are used as encoders in this framework. The CNN encoder is used to encode images and the RNN encoder is used to encode text. The neural network model is the decoder that merges both the inputs from the encoders. The availability of datasets for creating these models is significantly greater in English than in other foreign languages. Some of the English language data that are used are Flickr8k , Flickr30k, MSCOCO . Because of the scarcity of data in the field of foreign languages, there has been less research in this area. German (Elliott et al.; 2016), Chinese (Lin et al.; 2014), and Japanese (Yoshikawa et al.; 2017) are just a few of the foreign languages in which researchers have used crowdsourcing to gather data for their studies. The monolingual image captioning model was developed by the authors to be used in future research.

1.2 Motivation

Image captioning in both English and native language has a wide range of potential applications. Image captioning can be used to organize and categorize photos stored on your mobile devices and personal computers. Image captions in the native language can be used to describe millions of images on social media platforms, which are becoming increasingly popular. It is possible for satellites to use well-trained image caption models to describe scene features in a disaster-affected area, in order to alert the appropriate authorities. It is possible to use the native image caption model on an e-commerce website that is tailored to a specific region. It can also be used to provide audio descriptions of images for people who are visually impaired.

1.3 Research Question

Chinese, Spanish, and English are the three most popular languages in which extensive image caption research is done. The amount of research done for the Hindi- language is very low which is the fourth most spoken language in the world. The language is widely spoken throughout the Indian subcontinent, and the vast majority of the population in that region is unable to communicate effectively in English. As a result, image captioning in the Hindi language can be extremely beneficial for people who do not speak any other language other than Hindi. In response to this problem, the question “**How Can We Generate Semantically Correct Image captions in Hindi Using Deep learning Models**” is posed. The Flickr8k dataset is chosen as the dataset for this research because it is the most portable dataset for image captioning considering the limited computational power of the system. The main objective of this research is to generate Hindi captions. To achieve this objective it is important to have image captions in Hindi. The Hindi descriptions dataset version of Flickr8k is made available by Rathi (2020). The dataset contains Hindi captions that have been machine-translated using the Google Cloud Translator API.

1.4 Proposed Implementation Method

This research proposes an encoder-decoder model for the generation of Hindi image captions. This encoder-decoder model will make use of both unidirectional and bidirectional LSTMs. Human and machine evaluation is used to assess the quality of the caption produced by this neural network model, which was trained with these datasets. For machine evaluation BLUE (Bilingual Evaluation Understudy) score is used Papineni et al. (2002). To the best of my knowledge, the proposed technique for Hindi image description has not been tested on the chosen dataset. The proposed research work can be used as a baseline for this dataset on Hindi image captioning.

The paper is divided into different sections. Section 2 contains the literature review of the work done in image captioning for English and Foreign Language. Section 3 discusses the methodology that was used to generate image captions in conjunction with how the dataset was obtained. Section 4 explains about the model architecture in detail. Section 5 discusses the environmental setup on which the models were trained and how the models were trained to generate Hindi image captions. Section 6 delves into the experimental findings and brief discussion on the findings of the project. Section 7 concludes the work and looks at the prospects for future research.

2 Related Work

This section discusses various image captioning work done in English and foreign languages by various authors. The section is divided into subsections based on the language used for image captioning. These subsections also discuss and analyze the various methodologies and deep learning techniques used to obtain image captioning results. Subsection 2.1 discusses about image captioning done in English language and subsection 2.2 about Image captioning in foreign language.

2.1 Image Captioning In English Language

Over the last decade, many research groups and authors have worked on image captioning and achieved significant results with their models. These image captioning studies have been conducted for the English language-based models due to the availability of various English language datasets. The earlier work in image caption was based on two methods sentence-template-based method and the retrieval-based method. These methods were not flexible enough as they were dependent on hard-coded structures (Bai and An; 2018). As time passed, these methods became extinct, and modern Deep Neural Network (DNN) methods were adopted. DNN has produced superior results in the fields of Computer Vision (C.V) and Natural Language Processing (NLP) when compared to traditional methods. Many DNN approaches were adopted as interest in Image Captioning approaches has grown, and these DNN methods are discussed in this section.

2.1.1 Encoder- Decoder Approach

The encoder-Decoder framework approach is one of the common Image captioning approaches as it generates semantically correct and meaningful captions. This framework was first implemented by Kiros et al. (2014). The framework's encoder ranked images and sentences while the decoder was to generate image captions from scratch. This framework used LSTM to encode sentences. Multimodal RNN was used with CNN by Mao et al. (2014). Vinyals et al. (2015) improved upon this model by making use of LSTM to generate the caption and achieved the state of the art results on the benchmark Pascal dataset with a BLUE score of 59. Because the visual information was only provided at the beginning of the process, there is a disadvantage to this process that can be minimized. Donahue et al. (2015) created a model that was similar to (Vinyals et al.; 2015), but Donahue et al. (2015) included visual information at each step of his process.

2.1.2 Semantic Embedding for Image Caption

The encoder-decoder approach can be improved by incorporating semantic embedding to improve the quality of the captions that are generated. According to Zhang et al. (2020), CNN is unable to accurately describe all elements of a scene in the generated text. The author used a text corpus to extract semantic information from a text to fix this issue. This was done to fill the gap between the text generated and the image data. Text information derived from the image's available textual cues can be used as semantic information and fed into the model to assist it in producing better text. Gupta and Jalal (2020) used the textual cue information in the image was combined with the image's global feature and fed into an LSTM to help generate more meaningful captions. When relevant textual cues to images were found, the captions generated were semantically

correct and meaningful. The unrelated text was generated when the textual cues and images were uncoordinated.

Most of these models use only unidirectional LSTM which stores data from the past. These limitations can be overcome by employing Bi-directional LSTM, which uses both future and past information and can deal with longer text sequences.

2.1.3 Bidirectional Approach

Bidirectional LSTM models outperform LSTM models because they store both past and future information about the text that is fed to them. Wang et al. (2016) developed an image captioning model using pre-trained CNN and Bidirectional LSTM. The BLUE1 score obtained for the used flickr8k data was 65.5, which was quite high, indicating that the quality of caption produced by this model was very good. Xiao et al. (2019) created a comparable model using DSEN (Densely Semantic Embedding Network) and Bi-LSTM. The Bi-LSTM model produced a BLUE1 score of 72.0. Bi-Lstm can be used to compare results with uni-directional LSTM because it retains more information and produces better captions. This research uses both bi-directional and uni-directional LSTM. The Bi-LSTM model still misses some of the features that are in the images and this limitation can be resolved by adding an attention layer to the image captioning model.

2.1.4 Attention Mechanism

The default encoder-decoder image caption model captures the image's global information while leaving out minor details, which can be redundant when describing an image. Using the attention mechanism layer to capture small features of the image and feed them to the LSTM at each step can solve this problem. The quality and meaning of the generated caption improve with these features. The Attention Mechanism simulates human brains that are capable of paying attention to image features. Xu et al. (2015) implemented the first attention model, which used soft and hard attention with LSTM on the Flickr8K, Flickr30k, and MS COCO datasets. They used the BLUE score to evaluate the results. Dang et al. (2019) used an attention mechanism with two pre-trained CNNs. The authors trained two models, one with and one without an attention mechanism. When tested, the results produced by the model using the attention mechanism were superior.

The English image caption models have been generated by using a range of methodologies. The three main methodologies are as follows: Multimodal architecture, Encoder-decoder framework, and Attention Mechanism.

2.2 Image captioning In foreign language

2.2.1 Image captioning in Japanese

As there is a lack of corpora in a foreign language for Image caption research, Miyazaki and Shimizu (2016) constructed a Japanese version of the MSCOCO English dataset. This dataset was dubbed the 'YJ caption 26k Dataset', and the captions for the images were obtained through crowdsourcing. Miyazaki and Shimizu (2016) represented the comparison of three learning techniques to best describe the Japanese language in their research. These three methods of learning were monolingual, alternative, and transfer. In a similar way, Yoshikawa et al. (2017) constructed a Japanese version of the

MSCOCO dataset named ‘STAIR captions’. These captions were also created with the assistance of crowdsourcing. The number of captions provided for the available images differs between the two datasets, ‘STAIR caption’ and ‘YJ caption 26k dataset’. ‘STAIR captions’ provided 820,310 Japanese captions for all images (164,062 images), while ‘YJ captions’ provided 131,740 Japanese captions for only 26,000 images. Yoshikawa et al. (2017) trained their model with machine-translated captions and the crowdsource captions. The researchers found out that captions generated from crowdsource captions generated fluent captions. Tsutsui and Crandall (2017) made use of ‘YJ 26k captions’ to develop their model. The dual-language model with English and Japanese model that was developed by the authors used artificial tokens at the start of the sentence They also trained the monolingual model with the Japanese language. After conducting extensive experiments and evaluating the model, they came to the conclusion that the monolingual model outperformed the dual-language model.

From the above-reviewed literature we can say that the monolingual model generates better captions when trained with crowd-sourced captions. Because crowd-sourced captions are more accurate than machine-translated captions, the accuracy and quality of crowd-sourced captions will always be superior to machine-translated captions.

2.2.2 Image captioning in Chinese

According to the literature review for Japanese captions, we discovered that crowdsourcing and machine translation are two methods of generating training captions for the preferred language. Another method of obtaining captions is through the use of the human translation of already existing English captions. Li et al. (2016) used the Flickr8k image dataset to generate Chinese captions. Using crowdsourcing, machine translation, and human translation, he created captions for these datasets. Flickr8k-CN was the name given to the crowd-sourced dataset. Li et al. (2016) came across a culture gap when he was crowd-sourcing the captions. The caption that was generated in English described a lady in the picture as an Asian woman, whereas the captions that were collected from the Chinese population through crowdsourcing described the same woman as a middle-aged lady. As a result of their findings, the researchers concluded that when image descriptions are crowd-sourced based on geographic location, there can be a cultural difference. Li et al. (2016) conducted experiments with all three types of captions and came to the conclusion that machine-translated captions outperformed human-translated captions, and that crowd-sourced captions generate the highest accuracy of any of the three caption types. Crowd-sourced captions are natural and fluent when compared to machine-translated captions. To bridge this gap Lan et al. (2017) proposed ”Fluency guided framework”. In this framework, the non-fluent machine-translated captions were edited in order to transform them into fluent machine-translated captions. The fluency-guided framework model-outperformed the machine-translated model. To support this research a Chinese version of the MSCOCO dataset was made by Li et al. (2019). This dataset was named “COCO-CN” and was constructed using crowdsourcing.

After reviewing the above experiments we can say that crowdsource caption generates better and more fluent captions than human-translated captions. However, both of these methods have a disadvantage in that they require a significant amount of time to collect the captions. In order to resolve this, we can make use of machine-translated captions, which we can then use to train our models. This research focuses on the development of an image captioning model that includes machine-translated descriptions.

2.2.3 Image captioning in European Language

Apart from English, Japanese, and Chinese, significant research has been conducted in some European languages such as French, Spanish, Dutch, and German. A multilingual image captioning approach was designed by Elliott et al. (2015), where text in both English and German is based against image features at the same time. The researchers utilized images from the IAPR-TC12 data source that were associated with the German captions to train the multilingual model. The dataset contains 20k images with descriptions both in English and German. The dataset was first introduced by Grubinger et al. (2007). Flickr8k dataset was used by Elliott et al. (2016) to build a German description dataset for image captioning research. Captions for the images in this dataset were generated in two ways: first, the English descriptions were translated into German using a professional translator, and second, crowd-sourced captions were collected for the images in the dataset. This dataset was given the name "Multi30k." In this research, the author concluded that human evaluation is a better metric than machine translation for evaluating image captions.

van Miltenburg et al. (2017) made an image caption model for Dutch language. The captions for this research were collected via crowdsourcing and merged with the Multi30k dataset. The Dutch description generated in this model was compared to the English and French description and the authors found out that the description varied because of cultural description. The results were similar to that obtained in Japanese (Yoshikawa et al.; 2017) and Chinese (Li et al.; 2016). The dataset was released by van Miltenburg et al. (2017) and named DIDEC(The Dutch Image Description and Eye-tracking Corpus). Verbalized image description model in Spanish was introduced by Gomez-Garay et al. (2018). Elliott et al. (2016) asserted that human evaluation is the most effective method of evaluation in the field of image captioning.

2.2.4 Image captioning in Hindi

In the field of Hindi Image captions, some work has been done, but on a variety of datasets. To train the model, the majority of these datasets use machine-translated text. Dhir et al. (2019) made use of a deep attention-based framework for their Hindi image captioning model. The MSCOCO dataset was used by the researchers, and the description was translated using Google Translate. The authors used two human annotators to manually correct and check the descriptions in order to maintain the quality of the descriptions. They obtained a BLUE1 score was 0.57. Mishra et al. (2021) used transformer networks to generate Hindi captions. The dataset for this experiment was created using Google Translate. For translation, the authors used the MS COCO dataset. The proposed model received a very respectable BLUE1 score of 62.5.

This section examines the literature on image captioning in both English and foreign. Deep learning models of various types have been applied to a wide range of datasets. There has been very little research into Hindi image captioning. The MSCOCO dataset served as the foundation for the majority of the models. Following an analysis of the techniques and dataset, the project has decided to use the Flickr8k machine-translated dataset and the encoder-decoder framework. The research will compare the results of uni-directional and bi-directional LSTM with various types of pre-trained CNN for image feature extraction. From the findings of the literature review, we can safely assume that human evaluation is the most effective method for evaluating the automatically generated

captions. The BLUE score will be used as a metric for machine evaluation in this research. As a metric in this study, both human and machine evaluations will be used. Computer vision and the natural language processing field can benefit from this research of Hindi Image captioning.

3 Methodology

In this section, we will go over the methodology that was used for this research. There are two methodologies that can be used: CRISP-DM (Cross-Industry Standard Process for Data Mining) and KDD (Knowledge Discovery in Database). The deployment layer distinguishes these two approaches. The deployment layer is provided by CRISP-DM. We will use the KDD approach because there will be no deployment in this research. The KDD approach involves five steps in its process: data collection, pre-processing, transformation, data mining, and evaluation.

3.1 Data Collection

Flickr8k dataset is used for this research. This dataset has 8000 images with 5 descriptions for each dataset(40,000 descriptions). As this research is for Hindi captioning we have made use of machine-translated text. The English description provided by this dataset was translated using google API. The dataset was already translated by using this process and made available at a GitHub repository by Rathi (2020).



Figure 1: Translated Captions Example

Four types of datasets were created by the author for experimentation. Several versions of clean and unclean datasets were created, with two datasets containing 5 image descriptions and two datasets containing 1 description per image. The unclean-5 sentences version of the dataset uses the raw translated sentences and no type of cleaning was done for that sentences. This research has made use of only unclean-5 sentence data.

3.2 Data Preprocessing and Transformation

This section explains how the data was pre processed and transformed before feeding it in the model to train.

3.2.1 Image Data

Before feeding image data into a deep neural network model, the data must first be converted into a vector format for processing. A survey of deep learning image captioning models was carried out by Hossain et al. (2019), and it was discovered that the majority of current state-of-the-art research relied on pre-trained CNN models with fixed-size vectors. For image caption generation various types of pre-trained CNN can be used. AlexNet, ResNet50, VGG16, VGG19, and InceptionV3 are some of the most commonly used pre-trained CNN. VGG16, ResNet50, and InceptionV3 CNN models are used in this study for transfer learning. When using the pre-trained CNN, the last layer of the CNN is removed because it is mostly used for image classification. In this study, the second last layer, which returns the image's features, was used. The extracted features from these CNNs were saved in a pickle file so that they could be reused without rerunning the process, saving time during model execution.

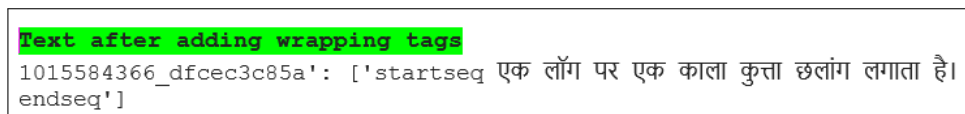
3.2.2 Text Data

(a) Text Cleaning:

The unclean-5 sentences of the Flickr8k-Hindi dataset has a vocabulary size of 8,652 and a description length of 39. Stop words were not removed from the dataset. removing stop words can lead to less fluent image description (Lan et al.; 2017). Punctuation and numbers were removed from the text during the pre-processing step.

(b) Wrapping Tags:

To teach the machine the start and end of a sentence, the text data was wrapped with a start and end marker. The start marker used is 'startseq' and the end marker used is 'endseq'. Figure 2 shows an example of how the sentences look after adding wrapping tags.



```
Text after adding wrapping tags
1015584366_dfcec3c85a': ['startseq एक लॉग पर एक काला कुत्ता छलांग लगाता है।
endseq']
```

Figure 2: Text after adding wrapping tags

(c) Tokenization:

Text cannot be directly processed by neural network models. The text should be converted into integer tokens to solve this problem. The pre-processed image description is converted into integer tokens in this study using Keras Tokenizer. The integer tokens are then converted into floating-point values. To keep the size of these vectors similar to the size of the image vector, zero padding is added. This enables the tokens to be easily merged and process as input to the model's decoder.

3.3 Evaluation Metrics

To evaluate the generated caption this research uses machine evaluation and human evaluation. For machine evaluation, the BLUE score is used and for human evaluation, a small sample of the caption was evaluated by a Hindi-speaking crowd.

3.3.1 BLUE

BLUE(Bilingual Evaluation Understudy) is used to evaluate machine-translated sentences. The closeness between the reference and candidate or predicted sentences can be measured with the help of a BLUE score. By counting n-gram co-occurrences, it assesses the quality of the generated description in the context of several reference descriptions. The higher the n-gram score, the more fluent the generated description is (Papineni et al.; 2002). Up to four n-grams are most commonly used. One to four n-grams are used in this research. BLEU-4 evaluates the fluency of the generated description, whereas BLEU-1 evaluates the adequacy of the description. Moreover, it evaluates the precision of the generated description, where precision is defined as a ratio between the number of overlapping words in the candidate sentence and the total number of words. One of the main reasons for using BLUE in this research was as it is language-independent.

3.3.2 Human Evaluation

Human evaluation is the best way to assess the quality of generated captions. The research is unable to evaluate the 2000 test image description due to time and resource constraints. A small sample of data from the best model with a high BLUE score is taken into account. A survey with a sample of images and descriptions was created and shared with a Hindi-speaking crowd for human evaluation. The quality of the image captions is graded on three levels: Good, Average, and Bad. When a caption is rated as ‘Good,’ it is fluent and grammatically correct, and it describes the majority of the image’s features. When there is a minor error in the generated captions, the captions are rated ‘Average.’ When the descriptions are irrelevant to the image, the captions are rated ‘Bad’.

3.4 Justification for Model selection

The study employs pre-trained CNN because they are simple to use, produce better results, and require less time to train. When compared to custom-built CNN, pre-trained CNN produces more accurate results. After extracting the image’s features, there is a problem with including the image’s scene in a generated text. The unidirectional LSTM is used to solve this problem. The generated text’s precision is limited to short sentences. To solve this limitation bi-directional LSTM is used. Bi-directional LSTM is good when using longer sentences as it retains more information.

4 Design Specification

In this section we will discuss how the encoder-decoder model was selected and designed. The process flow of the model will also be discussed.

4.1 Model Structure

Selecting a suitable structure for the encoder-decoder framework is very important. Tanti et al. (2018) in his research compared 16 different models and came to the conclusion that 'Merge model' is the best model architecture for encoder decoder framework. Figure 4 depicts the merge model. When creating the merged model, the word sequence vector

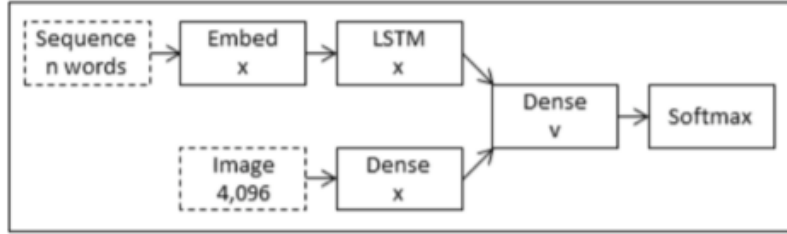


Figure 3: Merge Model

and the Image feature vector are both used as inputs. The decoder model then uses the combination of these two vector inputs to generate the next likely word in the sequence. In particular, the model performs better when text data and image data are dealt with exclusively. The best neural network model to deal with text data is RNN-LSTM (Tanti et al.; 2018). This research uses unidirectional and bidirectional LSTM to encode sentences and pre-trained CNNs to encode image data. These encoded outputs from image and text data are merged and there are various ways to encode these inputs, such as addition, multiplication, and concatenation. The addition is the best way to merge the encoded input (Tanti et al.; 2018). Hence, we have used addition to merge the image and text vector. The dense layer is the decoder which takes a combination of both image and text vectors. The model's final layer, the softmax layer, uses a greedy search to return a single word based on the probability of the previous word and the image feature that was generated.

4.2 Design Process Flow

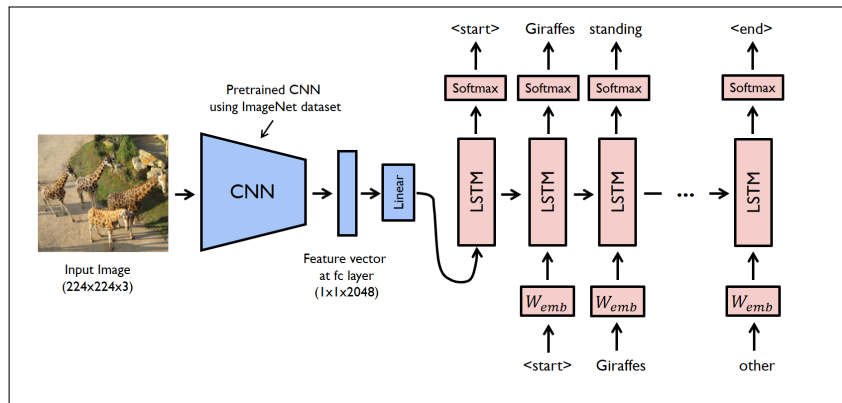


Figure 4: Process Flow of the Model

Figure 4¹ shows the process flow of an encoder-decoder model. The model is a combination of CNN-RNN models. Two inputs are given to the model to train, first, Image, second, corresponding captions. We assign one word to each LSTM layer, and each LSTM layer predicts the next word, which is how the LSTM model learns from captions and optimizes itself. Pre-trained CNN is used to extract image features. These features can be used directly to correlate captions and image features. The model generates captions for the final layer that are the maximum length of dataset captions. The size of the last layer is the length of the vocab.

5 Implementation

This section discusses about implementation of the project. Environmental setup with the steps involved in the training and tuning of the model are discussed.

5.1 Environmental Setup

Google colab is used to carry out the project's implementation. Colab provides a powerful GPU (Tesla P100) with up to 16GB of RAM. Python 3.7 was used for the code modeling. The model was run using the Keras library.

5.2 Training Model

Several models with different pre-trained CNNs were used to conduct multiple experiments in this study. In this section we will discuss the training model that generated best captions for our test Image.

5.2.1 Image Features

Of all the implemented pre-trained CNN models, VGG16 provided the best results. The pre-trained CNN was imported by using the Keras library. After that, the CNN was initialized and restructured. By removing the last layer of the CNN, it was restructured. The last layer was removed from the model because it is a classification layer. The images' target size is set to 224*224. The image pixels are saved in a numpy array before being reshaped for the model. Following these processes, image features are extracted. The extracted image features are saved in a pickle file so that they can be used later without rerunning the feature extraction process. VGG16 generates an image vector of array size 4096 based on its last layer which was provided as an input to the image encoder and this is the reason why the input shape of the image encoder and image feature is the same.

5.2.2 Text Features

The image description file is the first to be loaded into the system. The description file contains an image Id and a list of 5 captions that describe the image. This file was processed and saved into a dictionary, with the key corresponding to the image ID and the values corresponding to the image description. These pre-processed descriptions are later saved in a text file. These inputs are used as a second input by the text encoder. The input sequence is expected to be 39 characters long, which is the maximum length

¹<https://medium.com/swlh/automatic-image-captioning-using-deep-learning-5e899c127387>

of the description. The input sequence is then processed by the encoding layer, which is followed by the dropout layer, which has a dropout rate of 0.5. The model is trained and evaluated using two types of LSTM layers: unidirectional and bidirectional LSTM.

5.2.3 Merging Image and Text features

The decoder takes inputs from both the image and text encoders. The add operation is used to combine both encoders. This is then sent to a dense layer that employs 256 neurons for both unidirectional LSTM bidirectional LSTM. Softmax is used as an activator in the dense layer. After completing all of the steps, the model is compiled with Adam as the optimizer.

5.2.4 Progressive Loading of Data

As the size of our image dataset is huge it needs a lot of memory to process. The system on which this model is run lacks sufficient memory to process the data as a whole. To address this issue, the data is loaded progressively in the model. A generator function is written that generates a single batch of datasets. While fitting the model, these batches of the dataset are passed to it. Keras supports progressive loading of data.

5.3 Tuning the Model

The models in this study were manually tuned. Epochs and steps per epoch were taken into account as hyperparameters. On a small sample of datasets, the epochs were changed from 5 to 20. On 10 and 20 epochs, the model performed better. The length of the train description was taken into account for steps per epoch. The final models for the research were trained on these parameters.

6 Evaluation

6.1 Experiment -1 CNN-VGG16

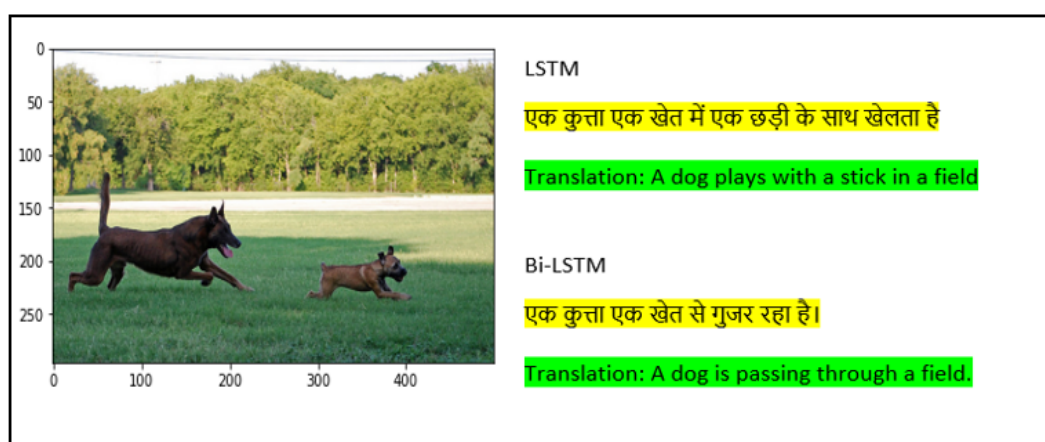


Figure 5: Captions Generated For VGG16 Model

The first experiment was carried out with VGG16 as the pre-trained CNN for feature extraction. Both LSTM and Bi-LSTM were used for text encoding. The models were designed to run on a range of different epochs. The best BLUE1 score for LSTM was 0.56, and the best BLUE1 score for Bi-LSTM was 0.58. The BLUE score did not differ significantly, but it was fairly obvious from the captions that the Bi-LSTM model yielded a better description for the image.

From the figure 5 we can see that both the LSTM models were able to identify the dog and field in the image. The LSTM model also identified an unwanted object—a stick—that was not in the image. The Bi-LSTM model did not identify any unwanted objects that are present in the image. When the context of the image is considered, the Bi-LSTM model caption makes more sense.

6.2 Experiment - 2 CNN- ResNet50

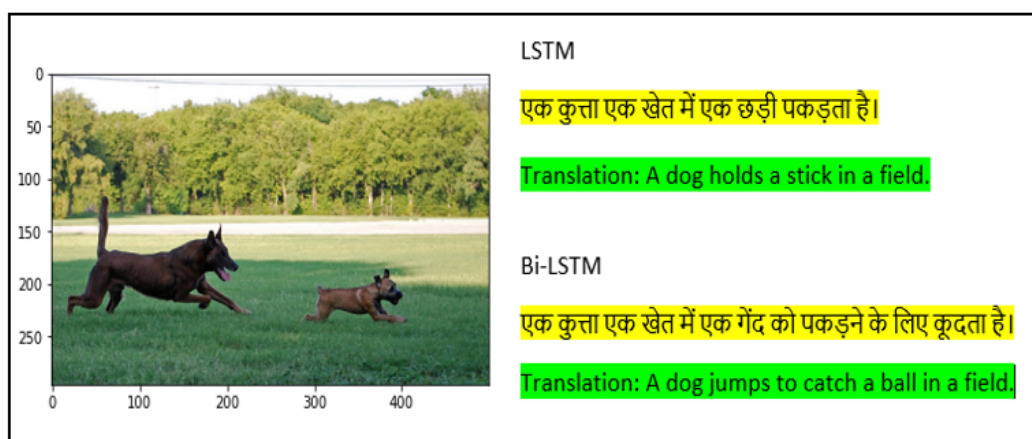


Figure 6: Captions Generated For ResNet50 model

In the second experiment, a ResNet50 pre-trained CNN was used to extract image features. Even for this model, LSTM and Bi-LSTM were used for text encoding. The experiments were designed to run for various number of epochs. The LSTM model yielded a BLUE1 score of 0.57, while the Bi-LSTM model yielded a BLUE1 score of 0.53. The BLUE score difference in both models was not significant, as it was in the VGG16 model. Even with this model, both models generate a caption that identifies the dog and the field in the image as you can see in figure 6. Both LSTM models predict an unwanted object that is not in the image, but the Bi-LSTM model also predicts an action, such as jumping in the captions, which adds context to the image.

6.3 Experiment - 3 CNN- InceptionV3

The third experiment used InceptionV3 to extract the features of the image. This model only used LSTM as encoder. The BLUE1 score generated for this model was 0.39 which is very low. From the figure 7 we can see that the captions generated are irrelevant. The sentence formation is incoherent and grammatically do not make sense. The model is not able to capture features or details from the image and this experiment was rejected based on these reasons.

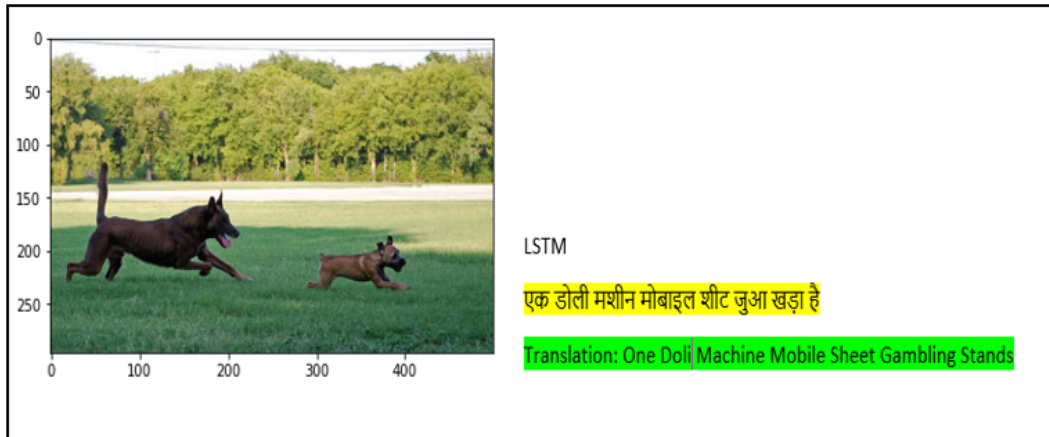


Figure 7: Captions Generated For InceptionV3 model

6.4 Human Evaluation

A survey² form with ten sample images and captions generated from the best model was created for this study. This survey was distributed to a Hindi-speaking audience. The image was rated based on the context and grammatical structure of the captions, which were rated as 'Good,' 'Average,' and 'Bad. Their responses were gathered, and the majority response to the images was taken into account. This survey had 21 respondents. 40% of the captions in the survey were rated as good, 30% were rated as average and 30% as bad.

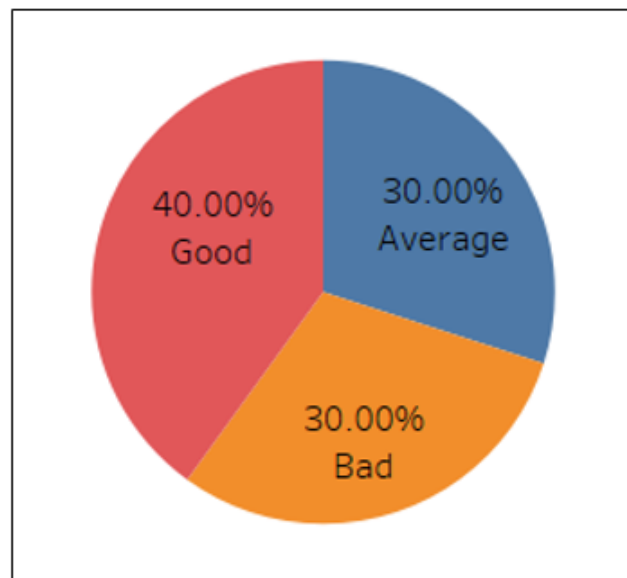


Figure 8: Survey Results

²<https://forms.gle/agw9duUSi21TxFcf8>

GOOD CAPTIONS: Image captions that described the features of the image accurately and were grammatically correct were categorised as good caption.



Figure 9: Good Captions

AVERAGE CAPTION: Image captions that do not describe all the images accurately and has some grammatical mistakes are considered as average captions.



Figure 10: Average Captions

BAD CAPTION: Image captions that do not make sense and miss all the features in the image are categorized as bad captions.



Figure 11: Bad Captions

6.5 BLUE Score of All the Experiments

Table 1 shows the BLUE score of all the experiments that were carried out. The vgg16 Bi-LSTM model achieved the highest BLUE1 score of 0.583 and the InceptionV3 model

with CNN had the lowest score of 0.39.

Table 1: BLUE SCORE TABLE

Dataset	Model Type	CNN	EPOCH	BLUE SCORE			
				B1	B2	B3	B4
Hindi Dataset	CNN-LSTM	VGG16	20	0.56	0.38	0.26	0.13
		InceptionV3	10	0.39	0.24	0.14	0.05
		RestNet 50	20	0.57	0.4	0.27	0.13
		VGG16	10	0.52	0.35	0.24	0.11
		RestNet 50	10	0.57	0.39	0.27	0.13
	CNN-BiLSTM	VGG16	20	0.54	0.37	0.25	0.12
		VGG16	10	0.583	0.4	0.27	0.12
		RestNet 50	20	0.53	0.37	0.25	0.11
		RestNet 50	10	0.52	0.35	0.24	0.12

6.6 Discussion

This study demonstrates how encoder-decoder models can aid in the generation of Hindi image captions. The use of both LSTM and Bi-LSTM assists us in differentiating the level model performance. When compared to the LSTM model, the Bi-LSTM model produced better captions and provided a higher BLUE score. By staying true to its definition, the Bi-LSTM model has provided more contextually meaningful and informative sentences than the LSTM model.

According to the findings of this study, human assessment is the ideal tool for estimating the generated caption. The use of machine-translated captions is a limitation of this study. It is not necessary that machine translation of captions from one language to another is always accurate due to grammatical structure challenges. From the literature reviewed we can say that crowd-sourced data generates more accurate captions. This study was unable to collect data from the Hindi-speaking community due to time and resource constraints.

While analyzing the captions, we can see that the models identify unwanted objects in the image that are not present or repeat the same captions for the image after identifying a single feature such as a dog or a ball in the image. This is due to the fact that the model employs a greedy search. The model predicts the next most likely word based on the previously generated words and image features that have been collected. This method results in data overfitting. This disadvantage can be solved by utilizing an attention mechanism because it gives more weight to image features.

7 Conclusion and Future Work

7.1 Conclusion

The proposed techniques were successfully implemented on the Flickr8k-Hindi dataset. According to the results of the study, the Vgg16 Bi-LSTM model outperformed the LSTM models with other pre-trained CNN. The captions generated by this model were significantly better, with the highest BLUE1 score of 0.583. Human evaluation was

done was on the caption by performing a survey on a small sample of the images. The research was able to generate captions that were semantically correct and meaning full. the objectives of the research were achieved. This research model can be used as a guideline for future work.

7.2 Future Work

The limited work done in Hindi captioning leaves a lot of room for improvement. The model developed in this study can be applied to a variety of other datasets. Because there is no crowdsourced dataset available for Hindi Image captioning, a higher quality dataset can be used to push the model for more human-like performance. An attention model can be used to further improve the quality of the generated captions. Reinforcement learning can also be used to predict captions. The use of a capsule network can be taken into consideration to extract image features. This study only scratches the surface of the possibilities in Hindi image captioning; by employing the aforementioned methodologies and techniques, better models that address the limitations of this study can be developed.

Acknowledgement

I would like to express my gratitude to Professor Noel Cosgrave for his unwavering support and weekly feedback sessions. The feedback session has been extremely beneficial in that it has assisted me in improving my research. I'd also like to express my gratitude to my parents for their unwavering support throughout this master's program.

References

- Bai, S. and An, S. (2018). A survey on automatic image caption generation, *Neurocomputing* **311**: 291–304.
- Dang, T. X., Oh, A., Na, I.-S. and Kim, S.-H. (2019). The role of attention mechanism and multi-feature in image captioning, *Proceedings of the 3rd International Conference on Machine Learning and Soft Computing*, pp. 170–174.
- Dhir, R., Mishra, S. K., Saha, S. and Bhattacharyya, P. (2019). A deep attention based framework for image caption generation in hindi language, *Computación y Sistemas* **23**(3).
- Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K. and Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2625–2634.
- Elliott, D., Frank, S. and Hasler, E. (2015). Multilingual image description with neural sequence models, *arXiv preprint arXiv:1510.04709* .
- Elliott, D., Frank, S., Sima'an, K. and Specia, L. (2016). Multi30k: Multilingual english-german image descriptions, *arXiv preprint arXiv:1605.00459* .

- Gomez-Garay, A., Raducanu, B. and Salas, J. (2018). Dense captioning of natural scenes in spanish, *Mexican Conference on Pattern Recognition*, Springer, pp. 145–154.
- Grubinger, M., Clough, P., Hanbury, A. and Müller, H. (2007). Overview of the imageclef-photo 2007 photographic retrieval task, *Workshop of the Cross-Language Evaluation Forum for European Languages*, Springer, pp. 433–444.
- Gupta, N. and Jalal, A. S. (2020). Integration of textual cues for fine-grained image captioning using deep cnn and lstm, *Neural Computing and Applications* **32**(24): 17899–17908.
- Hossain, M. Z., Soheli, F., Shiratuddin, M. F. and Laga, H. (2019). A comprehensive survey of deep learning for image captioning, *ACM Computing Surveys (CSUR)* **51**(6): 1–36.
- Kiros, R., Salakhutdinov, R. and Zemel, R. S. (2014). Unifying visual-semantic embeddings with multimodal neural language models, *arXiv preprint arXiv:1411.2539*.
- Lan, W., Li, X. and Dong, J. (2017). Fluency-guided cross-lingual image captioning, *Proceedings of the 25th ACM international conference on Multimedia*, pp. 1549–1557.
- Li, X., Lan, W., Dong, J. and Liu, H. (2016). Adding chinese captions to images, *Proceedings of the 2016 ACM on international conference on multimedia retrieval*, pp. 271–275.
- Li, X., Xu, C., Wang, X., Lan, W., Jia, Z., Yang, G. and Xu, J. (2019). Coco-cn for cross-lingual image tagging, captioning, and retrieval, *IEEE Transactions on Multimedia* **21**(9): 2347–2360.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C. L. (2014). Microsoft coco: Common objects in context, *European conference on computer vision*, Springer, pp. 740–755.
- Mao, J., Xu, W., Yang, Y., Wang, J. and Yuille, A. L. (2014). Explain images with multimodal recurrent neural networks, *arXiv preprint arXiv:1410.1090*.
- Mishra, S. K., Dhir, R., Saha, S., Bhattacharyya, P. and Singh, A. K. (2021). Image captioning in hindi language using transformer networks, *Computers & Electrical Engineering* **92**: 107114.
- Miyazaki, T. and Shimizu, N. (2016). Cross-lingual image caption generation, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1780–1790.
- Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation, *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318.
- Rathi, A. (2020). Deep learning approach for image captioning in hindi language, *2020 International Conference on Computer, Electrical & Communication Engineering (IC-CECE)*, IEEE, pp. 1–8.
- Tanti, M., Gatt, A. and Camilleri, K. P. (2018). Where to put the image in an image caption generator, *Natural Language Engineering* **24**(3): 467–489.

- Tsutsui, S. and Crandall, D. (2017). Using artificial tokens to control languages for multilingual image caption generation, *arXiv preprint arXiv:1706.06275* .
- van Miltenburg, E., Elliott, D. and Vossen, P. (2017). Cross-linguistic differences and similarities in image descriptions, *arXiv preprint arXiv:1707.01736* .
- Vinyals, O., Toshev, A., Bengio, S. and Erhan, D. (2015). Show and tell: A neural image caption generator, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3156–3164.
- Wang, C., Yang, H., Bartz, C. and Meinel, C. (2016). Image captioning with deep bidirectional lstms, *Proceedings of the 24th ACM international conference on Multimedia*, pp. 988–997.
- Xiao, X., Wang, L., Ding, K., Xiang, S. and Pan, C. (2019). Dense semantic embedding network for image captioning, *Pattern Recognition* **90**: 285–296.
- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R. and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention, *International conference on machine learning*, PMLR, pp. 2048–2057.
- Yoshikawa, Y., Shigeto, Y. and Takeuchi, A. (2017). Stair captions: Constructing a large-scale japanese image caption dataset, *arXiv preprint arXiv:1705.00823* .
- Zhang, X., He, S., Song, X., Lau, R. W., Jiao, J. and Ye, Q. (2020). Image captioning via semantic element embedding, *Neurocomputing* **395**: 212–221.